Faster Interactive Segmentation of Identical-Class Objects With One Mask in High-Resolution Remotely Sensed Imagery

Zhili Zhang, *Associate Member, IEEE*, Jiabo Xu, *Student Member, IEEE*, Xiangyun Hu, Bingnan Yang, and Mi Zhang

***Abstract*— Interactive segmentation (IS) using minimal prompts like points and bounding boxes facilitates rapid image annotation, which is crucial for enhancing data-driven deep learning methods. Traditional IS methods, however, process only one target per interaction, leading to inefficiency when annotating multiple identical-class objects in remote sensing imagery (RSI). To address this issue, we present a new task—identical-class object detection (ICOD) for rapid IS in RSI. This task aims to only identify and detect all identical-class targets within an image, guided by a specific category target in the image with its mask. For this task, we propose an ICOD network (ICODet) with a two-stage object detection framework, which consists of a backbone, feature similarity analysis module (S3QFM), and an identical-class object detector. In particular, the S3QFM analyzes feature similarities from images and support objects at both feature-space and semantic levels, generating similarity maps. These maps are processed by a region proposal network (RPN) to extract target-level features, which are then refined through a simple feature comparison module and classified to precisely identify identical-class targets. To evaluate the effectiveness of this method, we construct two datasets for the ICOD task: one containing a diverse set of buildings and another containing multicategory RSI objects. Experimental results show that our method outperforms the compared methods on both datasets. This research introduces a new method for rapid IS of RSI and advances the development of fast interaction modes, offering significant practical value for data production and fundamental applications in the remote sensing community.**

***Index Terms*— Identical-class object detection (ICOD), image feature similarity analysis, interactive segmentation (IS), remote sensing images.**

I. Introduction

THE development of remote sensing technology has enabled the wide application of high-resolution remote sensing imagery (RSI), providing detailed views of the Earth's surface [1], [2], [3], [4]. Extracting valuable information from these images is crucial for applications in land cover classification, change detection, resource management, and disaster monitoring [5], [6], [7]. With the accumulation of extensive datasets, methods based on deep learning have achieved remarkable performance in the automatic processing of RSI tasks, yet their success heavily depends on the availability of high-quality annotated data [8], [9], [10]. The process of manual annotation is, however, time-consuming and costly, becoming a bottleneck in the field's development. Interactive segmentation (IS) technology [11] emerged as a solution, offering an efficient way to segment objects in images with minimal user input—such as clicking, scribbling, or boxes—significantly accelerating the generation of high-quality annotated samples.

Recent works [12], [13], [14] in deep learning-based IS technology have garnered significant attention for their ability to efficiently segment objects from images. IS methods segment objects from images using minimal prompts, such as marking a few points, drawing a box, or making scribbles on the target object as positive samples, and points outside the target object as negative samples. These prompts effectively guide machine learning algorithms to accurately segment objects of interest from images. Several approaches are highly efficient, requiring only a single point or box to accurately segment the target [15], [16]. All these methods, however, share a critical limitation: they can only segment one target at a time. This limitation is particularly significant in the context of remote sensing images, which often contain a large number of identical-class targets. In such cases, individually segmenting each target individually is time-consuming and inefficient, requiring the development of more advanced solutions that can handle multiple targets simultaneously.

To rapidly extract all identical-class targets in high-resolution RSI, we design a new and heuristic fast IS scheme, as shown in Fig. 1. This scheme first segments a building target by an IS model. Subsequently, the segmented target is used as a support one to retrieve and detect all identical-class ones within the image, marking them with boxes. Then, the IS model is applied for precise segmentation of all the boxed targets. Finally, interactive adjustments are made for the targets in the segmentation results that require optimization. The key

Received 6 October 2024; revised 18 November 2024; accepted 13 December 2024. Date of publication 19 December 2024; date of current version 30 December 2024. This work was supported in part by the Special Fund of Hubei Luojia Laboratory under Grant 220100028 and Grant 230700006; and in part by the Fundamental Research Funds for the Central Universities, China, under Grant 2042022dx0001. *(Zhili Zhang and Jiabo Xu contributed equally to this work.) (Corresponding author: Xiangyun Hu.)*

Zhili Zhang, Jiabo Xu, Bingnan Yang, and Mi Zhang are with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: zhangzhili@whu.edu.cn; xujiabo@whu.edu.cn; bingnan.yang@whu.edu.cn; mizhang@whu.edu.cn).

Xiangyun Hu is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China, and also with the Hubei Luojia Laboratory, Wuhan 430079, China (e-mail: huxy@whu.edu.cn).

Digital Object Identifier 10.1109/TGRS.2024.3520360

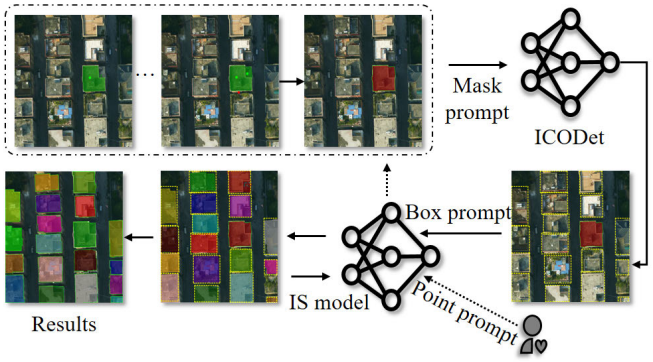


Fig. 1. Novel heuristic scheme for faster IS of all buildings simultaneously. The IS model represents the IS model. The proposed ICODet rapidly detects all objects of the identical class (highlighted with yellow dotted-line boxes) in images, guided by a support object with its mask.

innovation in our scheme lies in the introduction of a new task—identical class object detection (ICOD). The ICOD task is a subtask of the IS task designed for simultaneous IS of identical-class objects within an image. This task aims to query and detect all objects of the same class in an image based on a support object and its mask. It ingeniously leverages the characteristic of RSI, where geographically proximate identical-class objects typically exhibit visual similarity.

The proposed task is based on image feature similarity metrics, with similar ones, including image retrieval, few-shot object detection (FSOD), and object counting, as illustrated in Fig. 2. Image retrieval [17], [18] involves searching for similar images in a large image database based on text descriptions or reference images; FSOD [19], [20] identifies and locates objects in images under conditions of limited annotated samples; class-agnostic object counting [21], [22] involves calculating the number of specific type objects in images or video frames by exploiting the image self-similarity property. Compared to these tasks, the work in this article focuses only on querying all targets in the same category as the supporting targets and outputting their locations. It is worth noting that one-shot object detection [23], similar to our method, is a specialized case within FSOD. The approach, however, requires the model to be pretrained using samples from established base classes. Subsequently, to detect objects in new categories, the model needs only a support sample to accurately identify and locate objects from these previously unseen categories. Specifically, the ICOD task focuses on detecting only the objects that belong to the same class as the support target without detecting objects from other categories. It is specifically designed to address the challenges inherent in RSI, where multiple objects of the same class are present, significant variations exist across different regions, and there is a critical need for rapid IS. Through the comparative analysis presented in Table I, we demonstrate the distinctions among these tasks.

Based on this new sub-task, we propose a new identical-class object detection network (ICODet) for detecting objects of the same class at the regional/image level in RSI. The proposed network is designed by using similarity metrics-based deep features [24] and a two-stage object detection architecture [25]. First of all, we use convolutional neural networks (CNNs) [26] to extract high-level features from both an image and its support target. Subsequently, we design semantic and feature similarity analysis module (S3QFM) to separately analyze the similarities between the support target features and the queried image features in both semantic and spatial dimensions. The results from these similarity analyses are then fused to derive a comprehensive feature of similarity metrics. Following this, a region proposal network (RPN) [25] is employed to identify the preliminary positions and extract corresponding features of the queried targets. Finally,

TABLE I
DIFFERENCES AMONG FOUR TASKS USING IMAGE FEATURE SIMILARITY

Task	Level	Differences	Usages
Image Retrieval	Image	Focus on searching for images based on image or feature similarity.	Image library management, web searches, etc.
Few-Shot Object Detection	Object	Identify multiple category objects across images based on a limited number of cue category objects.	Object detection in sample-scarce scenarios, rapid adaptation to new objects, etc.
Class-Agnostic Object Counting	Object	Determine the number of specific type objects in images or video frames based on a prompt box or more for the same type object.	Crowd density estimation, traffic flow monitoring, agricultural resource management, etc.
Identical-Class Object Detection	Object	Identify the location of all objects of the same type in the image by the mask of the given object.	Image object annotation and sample generation.

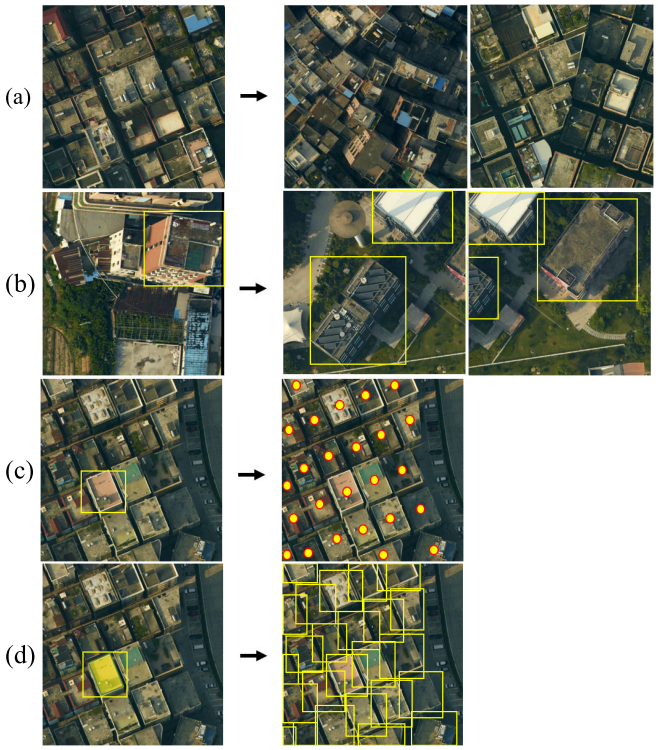


Fig. 2. Various tasks based on image feature similarity include: (a) image retrieval, (b) FSOD, (c) objection counting, (d) ICOD.

by applying a straightforward feature similarity comparison module (FSCM) between the features of the preliminary queried targets and the support features, the final query results are precisely obtained.

In this study, we focus on the ICOD task within the remote sensing scene. In response to this need, we have released a dataset containing buildings from six cities in China. The buildings in this dataset show significant differences in characteristics, making them highly suitable for the ICOD task. To address the limitation of the dataset covering only a single category, we further use the DOTA dataset [9] for supplementation. The reorganized data includes 12 different categories, aiming to expand the applicability of the target query task, which is crucial for validating the effectiveness of the proposed methods. The main contributions of this study are summarized as follows.

1) We introduce an ICOD task to build a novel IS scheme for RSI, effectively facilitating the rapid extraction of identical-class land objects within images.
2) A new ICODet is proposed, leveraging image similarities by using a comprehensive analysis of both semantic and spatial features across support target features and queried image features.
3) We have released the EVLab Building dataset, which includes buildings from six Chinese regions, as well as a reorganized dataset comprising multiple categories of RS targets.
4) Our proposed network has outperformed comparable methods on both datasets, demonstrating its efficiency and applicability.

The remainder of this article is organized as follows. Section II offers a brief review of related work. Section III details the proposed ICODet network, including the feature similarity analysis module among others. Experiments and analyses are discussed in Section IV. Finally, Section V concludes the study.

II. Related Work

In this section, we concisely review literature pertinent to our study, focusing on IS and image feature similarity metrics.

A. Interactive Segmentation

Before the advent of deep learning, IS methods [27], [28], [29], [30] mainly depended on low-level image features and optimization-based graphical models, which often led to poor performance and efficiency. The breakthrough success of deep learning in semantic segmentation inspired a new generation of IS techniques. These methods transform user interactions into click maps for model input, significantly improving accuracy and efficiency. The first deep learning-based IS method [31] revolutionized the approach by using distance maps derived from clicks in combination with images. Several methods, such as DEXTR [32], FCA-Net [33], and f-BRS [34], have targeted various IS efficiency improvements, focusing on elements like extreme point identification and optimization during inference. Recent works include RITM [14] and PGR-Net [35], which integrates previous segmentation results and distance maps into inputs, and PseudoClick, which employs an additional module to simulate annotator clicks. The recent methods [12], [13] focus on refining segmentation locally using lightweight modules and approaches like Interformer [15] and SAM [16] preprocess images with larger models, improving IS performance through efficient, lightweight modules. Additionally, the latest IS methods [36], [37], [38] have adopted new strategies and universal prompt encoders to enhance segmentation performance. For example, GPCIS [37] leverages click points information to frame the IS task as a pixel-level binary classification model based on Gaussian processes (GPs), thereby improving the quality of IS targets. SEEM [36] encodes diverse prompting for all types of segmentation tasks, enabling IS of everything everywhere at once. DINOv [38] introduces a general visual prompting framework for open-set segmentation and detection tasks, enhancing the segmentation capabilities for visual targets. Despite these advancements, a common limitation remains: the inability to segment multiple objects of the same class simultaneously, highlighting a potential area of our proposed task.

B. Image Feature Similarity Metrics

The evaluation of image feature similarity plays a crucial role in various applications [39], [40], [41], employing metrics such as Euclidean, Mahalanobis, cosine similarity, and Matusita distances. These methods are pivotal in deep metric learning [42], meta-learning [43], and few-shot learning [44]. For instance, Dong and Xing [45] used prototypical networks and feature similarity metrics for few-shot segmentation. Category-agnostic object counting [21], [22], [46], [47] has, moreover, used analysis of feature similarity at different levels to query the number of targets. Image-based image retrieval [17], [48], [49], [50] primarily relies on the measurement of image feature similarity. FSOD [20], [51], [52], [53] based on meta-learning has shown the effectiveness of comparing feature similarity between support objects and query images, enabling detection with minimal annotated data. Recent FSOD works [54], [55], [56] improve detection performance by focusing on hard samples or class-agnostic metrics. For example, Yan et al. [54] introduced confusing proposals separation (CPS) and affinity-driven gradient relaxation (ADGR) to address missing annotations treated as background. Liu et al. [55] designed a novel FSOD approach for remote-sensing images that addresses labeling inconsistencies and improves proposal quality. Additionally, to mitigate bias toward novel classes from base class training, Han et al. [56] propose a class-agnostic aggregation method, which aggregates query and support features regardless of their categories. FM-FSOD [57] is based on the DETR framework and integrates large language models to achieve FSOD.

III. Proposed ICODet Network

This section first clearly defines our proposed ICOD task and subsequently provides an in-depth description of the proposed ICODet for RSI IS. Fig. 3 illustrates the framework

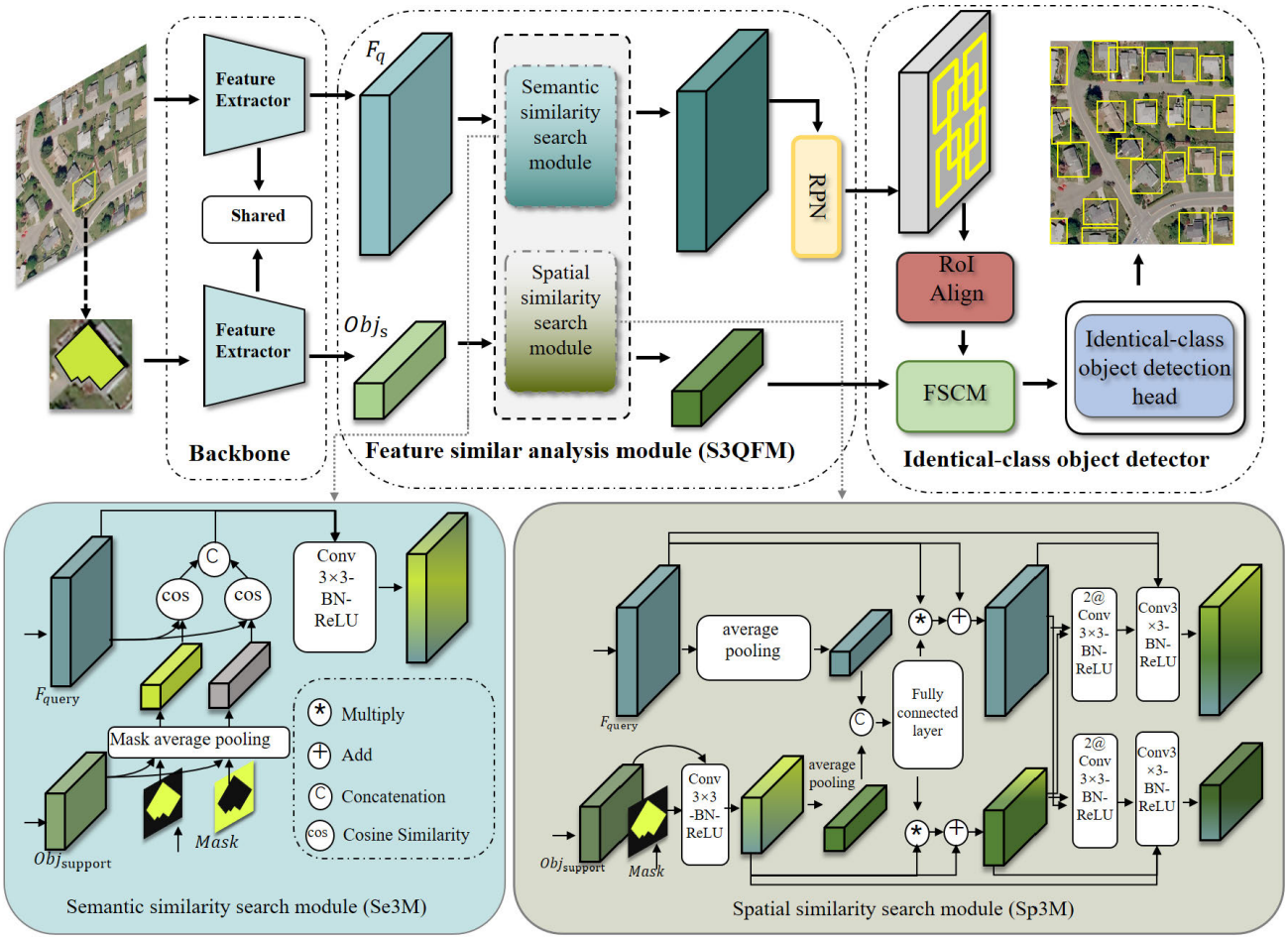


Fig. 3. Framework of the proposed ICODet for remote sensing image IS. FSCM.

of ICODet, comprising three principal components: the backbone, feature similarity analysis module, and identical-class object detector.

A. Task Definition

In ICOD, all targets of the identical category as the guided target are detected simultaneously. Specifically, for an image of a specific region containing N (where $N \geq 3$) targets of a certain class, the proposed task uses the mask of K (e.g., $K = 1, 2, 3$, default 1) targets to simultaneously query and locate the remaining $N - K$ class targets in the image, and uses a bounding box to represent the queried targets. Significantly, this task is limited to detecting objects of the same class as the support target and does not involve detecting objects of other classes. Both the support target and the objects to be detected are, moreover, located within the same image. The ICOD aims to efficiently detect identical-class targets in a specific region for fast IS of RSI.

B. Backbone

In the proposed ICODet, we employ a lightweight backbone, such as ResNet18 without its classification layer, to extract image features from both query and support target images. To get the support target image, we clip it from the input image using a mask's extended box, applying thresholds (e.g., 10, 15, or 20 pixels) to ensure a broader coverage of the target area. This clipped image is then resized to 224×224 pixels for compatibility with the backbone's input. The same backbone processes both the query and target images, producing feature maps at 1/16 of the original input size. Additionally, the target image mask is resized to match the size of the target features, serving as input for the feature similarity analysis module.

C. Feature Similarity Analysis Module

The S3QFM consists of two key components: the semantic similarity search module (Se3M) and the spatial similarity search module (Sp3M), as illustrated in Fig. 3. S3QFM leverages deep image features, along with the support object's features and mask, to learn the similarity between the image and the support objects from both semantic and spatial perspectives. This is the critical module for identifying identical-class objects.

First, Se3M operates by generating prototype features from the support mask and features, then matching them against the query image features to produce semantic-based search results. Concurrently, Sp3M uses both the mask and the features of the support and query images to perform matching and fusion across both global and local dimensions within the spatial

feature space. Finally, the process of S3QFM for the query image features ($\mathbf{F_q}$), the support target features ($\mathbf{F_s}$), and the mask ($\mathbf{M}$) can be formulated as follows:

$$\mathbf{FFM} = \mathrm{Cat}\big(\mathrm{Se3M}\big(\mathbf{F_q}, \mathbf{F_s}, \mathbf{M}\big), \mathrm{Sp3M}\big(\mathbf{F_q}, \mathbf{F_s}, \mathbf{M}\big)\big) \quad (1)$$

$$\mathrm{CBR}(x) = \mathrm{Conv3} \times 3(\mathrm{BN}(\mathrm{ReLU}(x))) \quad (2)$$

$$\mathrm{S3QFM}\big(\mathbf{F_q}, \mathbf{F_s}, \mathbf{M}\big) = \mathrm{CBR}(\mathrm{CBR}(\mathbf{FFM})) \quad (3)$$

where the abbreviations of "ReLU," "BN," and "Conv 3 × 3" correspond to the rectified linear unit, batch normalization, and a convolutional layer with a 3 × 3 kernel size, respectively. **FFM** represents the concatenation of query image features and support target features, along with their corresponding masks, after processing through Se3M and Sp3M. "Cat" denotes the concatenation operation. CBR(∗) refers to the sequence of operations comprising ReLU, BN, and a 3 × 3 convolution applied to the input variables.

For the Se3M, the given mask is first manipulated to extract foreground and background maps. These maps are then integrated with the support features via the masked average pooling (MAP) operation to generate the support feature prototypes. It can be computed by the following equation:

$$\mathbf{r_s} = \mathrm{MAP}(\mathbf{F_s}, \mathbf{M}) \quad (4)$$

where $\mathbf{F_s}$ represents the support target features, while $\mathbf{M}$ denotes the mask associated with the support target. The term $\mathbf{r_s}$ refers to the prototype features for the support target, encompassing both the foreground and background areas in the supporting image.

These prototypes are used for the generation of the matching results with the query features by using the cosine similarity metric, the ouput can be denoted as follows:

$$\mathrm{Similarity}_{\mathbf{fg}} = \mathrm{Cosine}\big(\mathbf{F_q}, \mathbf{r_{fg}}\big) \quad (5)$$

$$\mathrm{Similarity}_{\mathbf{bg}} = \mathrm{Cosine}\big(\mathbf{F_q}, \mathbf{r_{bg}}\big) \quad (6)$$

$$M\big(\mathbf{F_q}, \mathbf{r_s}\big) = \mathrm{Cat}\big(\mathrm{Simlarity}_{\mathbf{fg}}, \mathbf{Simlarity}_{\mathbf{bg}}\big) \quad (7)$$

where $\mathbf{r_{fg}}$ represents the prototype features of the foreground in the support image, while $\mathbf{r_{bg}}$ denotes the background prototype features for the support image. $\mathbf{F_q}$ refers to the query image features. $\mathrm{Similarity}_{\mathbf{fg}}$ represents the similarity map between the foreground prototype features and the query image features, while $\mathrm{Similarity}_{\mathbf{bg}}$ denotes the similarity map between the background prototype features and the query image features. $M(\mathbf{F_q}, \mathbf{r_s})$ indicates the similarity between the support image features and the query image features.

Finally, the matched results are concatenated with the query image feature for the input of a CBR(∗) function to produce the results of Se3M. This process generates similarity features based on learned semantic context.

The Sp3M achieves matching outcomes within the spatial feature dimension by performing feature fusion and matching at both global and local levels. Initially, it integrates support object features with its corresponding mask according to (2), denoted as $\mathbf{F_{sm}}$. The $\mathbf{F_{sm}}$ represents the support object features, which integrate learnable features from the target mask. Next, $\mathbf{F_{sm}}$ and the query image features undergo average pooling to obtain the target prototype features and the global features of the query image, respectively. These two feature vectors are concatenated and then passed through two fully connected layers (FCLs) to learn the common global weights (CGWs). The CGW is fused features based on the support feature vectors and the global features of the query image, designed to learn a correlated global representation of both feature types. The FCL can be expressed as follows:

$$\mathrm{FC}(x) = \mathbf{W}x + \mathbf{b} \quad (8)$$

$$\mathrm{FCL}(x) = \mathrm{Sigmoid}(\mathrm{BN}(\mathrm{FC}(\mathrm{ReLU}(\mathrm{BN}(\mathrm{FC}(x)))))) \quad (9)$$

where the "**W**" represents the parameters of the FCL, and "**b**" denotes the bias term.

After processing through the FCL(x), the global weight feature channels remain consistent with the input size, with dimensions of 1 × 1. These global weights are then applied to both the query and support image features through multiplication and addition, respectively, to produce features enhanced by global information.

In order to promote bidirectional interaction and matching between the support object features and query image features at a local scale, enhanced features are obtained through two merging operations. Note that after integrating the global weight information, both features are resized back to the original feature map size, which is 1/16 of the input image size. First, for the initial fusion of the query image features, the support image features are resized to match the query image features' dimensions and concatenated with them. This concatenated result is then fused using two CBR(∗) fuctions. Similarly, for the support object features, the query image features are resized to match the dimensions of the support object features and concatenated, followed by fusion using two CBR(∗) fuctions to obtain the enhanced support object features. Second, for the second fusion of the support object features and query image features, a CBR(∗) fuction is used to integrate the input features, the globally enhanced features, and the initially fused features of both types. This process yields the spatial-level matching results for the support object and query image. The support and query features enhanced by Sp3M maintain the same dimensionality as the input features.

D. Identical-Class Object Detector

Building on the architecture of Faster R-CNN for object detection, our proposed ICODet identifies positions of identical class objects. The model generates proposals by using fused features obtained from the S3QFM. In addition, a straightforward FSCM is implemented to compare the region of interest (RoI) features of each proposal with the enhanced support features, facilitating the extraction of finely compared features. This process culminates with the application of a detection head for bounding box (bbox) refinement.

Within the FSCM, we employ a straightforward yet effective method for feature comparison by calculating the difference between the support object features and each preliminary query object feature, followed by their fusion. Initially, the FSCM computes the difference between the support object features and the preliminary query object features by performing a

matrix subtraction to obtain the differential features of the two targets. Since these differential features enhance the distinct characteristics of different categories, it is essential to retain the original support object features and query image features as well. These differential features are then concatenated with the original support and query features. This combined features is subsequently fused using two convolutional layers, each with a kernel size of 3×3, to effectively integrate the three types of features. The overall process can be represented as follows:

$$\mathbf{FSCM} = \mathrm{CBR}\big(\mathrm{CBR}\big(\mathrm{Cat}\big(\mathbf{F_{so}}, \mathbf{F_{qo}}, \mathbf{F_{so}} - \mathbf{F_{qo}}\big)\big)\big) \tag{10}$$

where $\mathbf{F_{so}}$, $\mathbf{F_{qo}}$ correspond to the support object features and preliminary query object features.

Finally, the fused features obtained from the FSCM are fed into a classifier and a bounding box regressor. The classifier is used to determine whether the detected box contains an object of the same category as the support target, while the regressor is used to predict the positional parameters of these objects.

E. Loss Fuction

We optimized the loss function used in our proposed method by referencing the design of Faster R-CNN. Given that the task of ICOD is to detect targets of the same category as the support target, the adopted loss function does not include the classification of object categories. The overall loss function is composed of several distinct components, each designed to address specific aspects of the detection process.

The RPN Loss (L_{rpn}) is applied to the outputs from the RPN. This part of the loss function is critical for generating effective region proposals. It can be defined as follows:

$$L_{\mathrm{rpn}} = \lambda_1 \sum_i L_{\mathrm{cls}}\big(p_i, p_i^*\big) + \lambda_2 \sum_i p_i^* L_{\mathrm{mse}}\big(b_i, b_i^*\big) \tag{11}$$

where p_i is the probability of anchor i being an object, p_i^* is the ground truth, b_i is the predicted box, b_i^* is the ground truth box, L_{cls} is a classification loss applied to predict whether an anchor is an object or background, and L_{mse} is the mean squared error (mse) used to adjust the anchor positions to better fit the detected objects.

This objectness loss (L_{obj}) assesses the likelihood of each proposal containing an object after the feature comparison, which is crucial for determining the presence of targets. It is given by the following equation:

$$L_{\mathrm{obj}} = \sum_i L_{\mathrm{bin}}\big(o_i, o_i^*\big) \tag{12}$$

where L_{bin} is the binary cross-entropy loss, o_i is the predicted objectness score for each proposal, and o_i^* is the binary truth (1 if an object is present, 0 otherwise).

The mse loss further refines the bounding boxes predicted by the network. It can be denoted as follows:

$$L_{\mathrm{loc}} = \sum_{i \in \mathrm{pos}} L_{\mathrm{mse}}\big(t_i, t_i^*\big) \tag{13}$$

where only positive proposals (i.e., those containing an object) are considered. The final loss functions are as follows:

$$L_{\mathrm{det}} = L_{\mathrm{rpn}} + L_{\mathrm{obj}} + L_{\mathrm{loc}}. \tag{14}$$

TABLE II
DETAILS OF EVLAB BUILDING DATASET. "N.TR" MEANS THE NUMBER OF TRAINING SETS AND "N.TE" MEANS THE NUMBER OF TEST SETS. "TO." MEANS THE TOTAL NUMBER

No.	Region	Source	Resolution (m/pix)	N.Tr	N.Te
1	Taiwan	satellite	0.5	3000	638
2	Guangdong	aerial/satellite	0.3/0.5	3000	831
3	Chongqing	satellite	0.5	2894	395
4	Zhengzhou	aerial	0.1	1008	635
5	Wuhan	satellite	0.5	828	503
6	Xian	satellite	0.5	940	451
To.	——	——	——	11670	3453

IV. EXPERIMENTS AND ANALYSIS

A. Dataset Description

EVLab building dataset is carefully collected and manually annotated. It features a diverse range of buildings across six cities in China: Taiwan, Guangdong, Chongqing, Zhengzhou, Wuhan, and Xi'an, as illustrated in Fig. 4. Derived from Google Earth images and aerial images in 2019, this dataset is mainly used for the ICOD task in this article. The acquired images and their corresponding vector labels were cropped into 512×512 pixel resolution, using a 50% overlap between each cropped sample to maintain data coherence. The cropped vector labels were converted into polygon formats within the image coordinate system and saved as JSON files. To meet the ICOD task, we adopted the following criteria to preprocess the constructed dataset: first, we removed the buildings with less than 50 pixels in a sample; Second, if the number of buildings in a sample is less than three, the sample is excluded. After these steps, the dataset finally consists of 11 670 training samples and 3453 test samples, as shown in Table II.

In addition, we introduce two metrics for each sample to describe building distribution density across regions: the count of buildings (CoBs) and the ratio of the building area (RBA). CoB refers to the CoBs for each image, while RBA denotes the ratio of the total area of all buildings to the total pixel number of pixels per image. Fig. 5 reveals a concentration of buildings primarily on the left, indicating most samples have fewer than 30 buildings, suggesting low density. The top left part has numerous points, denoting many large-area building samples. Fewer points in the bottom left suggest sparsity. In particular, Guangdong samples frequently appear in the upper right, signaling higher-density building samples in this region

$$\mathrm{RBA} = \sum_{i=1}^{N} A_i \Big/ A_{\mathrm{image}} \tag{15}$$

where N is the number of buildings detected in the image, A_i is the area of the ith building and A_{image} the area of the image.

DOTA dataset [9] is further used for our work due to its inclusion of various types of ground objects. The dataset

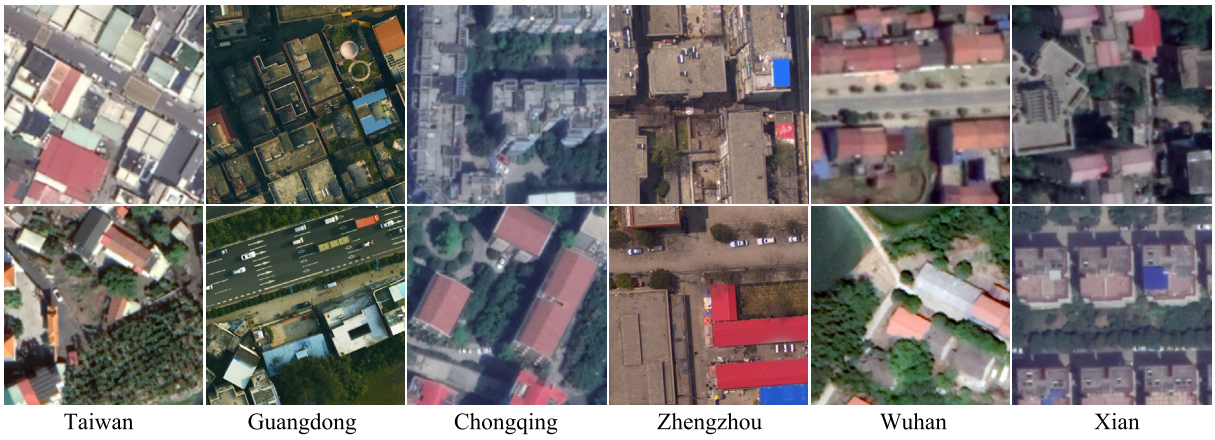


Fig. 4. Typical building samples of EVLab building dataset.

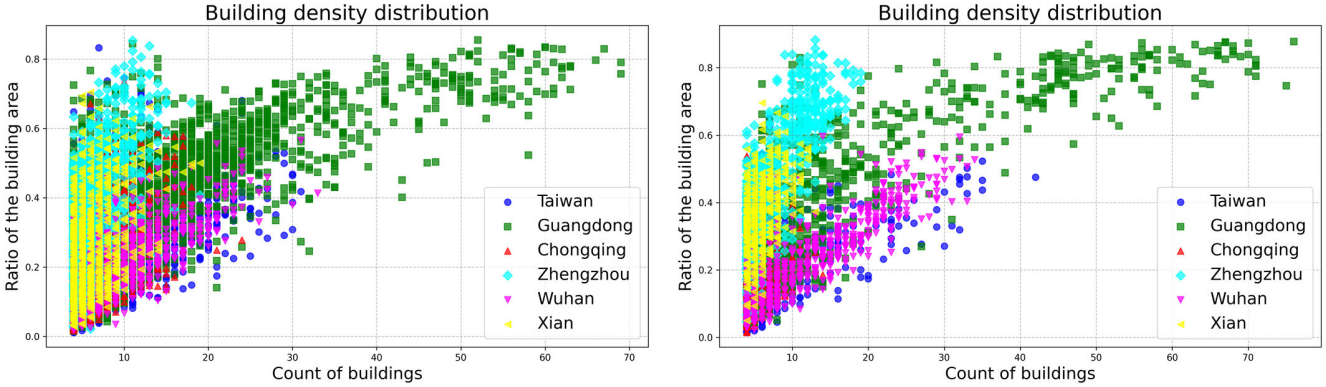


Fig. 5. Building density distribution of six regions in EVLab building dataset. From left to right: training and test sets.

is a large-scale, high-resolution remote sensing image object detection dataset. It comprises 18 different categories, totaling 11 268 images and 1 793 658 annotated instances. The resolution of the images ranges from 800 × 800 to 20 000 × 20 000 pixels, covering targets of various sizes, orientations, and shapes. In the DOTA dataset, each target instance is marked with an arbitrary quadrilateral. To fit our task, we performed a series of reprocessing steps: 1) images are cropped to 512 × 512 pixels; 2) each sample contains at least one type of ground object and the number of such types is more than 3; and 3) the number of training samples per category exceeds 50. In addition, target instances are annotated using an external rectangular box, while retaining its original quadrilateral annotation. After these processes, the data statistics are presented as shown in Table III.

TABLE III

DETAILS ABOUT THE PROCESSED DOTA DATASET. "N.TR" MEANS THE NUMBER OF TRAINING SETS AND "N.TE" MEANS THE NUMBER OF TEST SETS. "TO." MEANS THE TOTAL NUMBER

No.	Class name	N.Tr	N.Te
1	small-vehicle	3000	900
2	ship	2742	870
3	large-vehicle	2979	710
4	plane	1406	475
5	harbor	1197	433
6	storage-tank	801	314
7	tennis-court	752	204
8	swimming-pool	339	134
9	bridge	108	26
10	basketball-court	88	18
11	helicopter	57	8
12	baseball-diamond	48	16
To.	——	13517	4108

B. Experimental Settings

1) Evaluation Criteria: In our proposed ICOD task, bounding boxes are used to represent the results of querying targets of the same category in an image. To evaluate detection accuracy, we use the average precision (AP) metric, a standard measure in object detection. AP quantifies the trade-off between precision and recall and is calculated as the area under the precision-recall curve. This calculation relies on the intersection over union (IoU) metric, which assesses the overlap between predicted and ground truth bounding boxes. The IoU is defined as follows:

$$\mathrm{IoU} = \frac{B_{\mathrm{p}} \cap B_{\mathrm{gt}}}{B_{\mathrm{p}} \cup B_{\mathrm{gt}}} \tag{16}$$

where B_p is the predicted bounding box and B_{gt} is the ground truth bounding box.

We calculate the AP using the IoU metric. It represents the average over multiple threshold conditions, with IoU ranging from 0.50 to 0.95, in steps of 0.05. The AP is expressed as follows:

$$\mathrm{AP} = \frac{\mathrm{AP}_{50} + \mathrm{AP}_{55} + \cdots + \mathrm{AP}_{95}}{10}. \tag{17}$$

For the IS task, we set the IoU at 0.5 to deem a target detection correct for all experiments. We set the target score threshold to 0.65 because remote sensing targets are easily distinguishable from the background. Based on these settings, we use precision, recall, and F1 score for evaluation. For each target detected in an image, true positive (TP) represents the correctly predicted targets, false positive (FP) represents the incorrectly predicted targets, and false negative (FN) represents the targets that were not detected; therefore, the formulas for precision, recall, and F1 score are in the following equations. At last, we calculate the detection accuracy for each image and average all the results to serve as the metric for evaluating all methods

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{18}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{19}$$

$$\text{F1-score} = \frac{2 \times \mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}. \tag{20}$$

2) Parameter Setup: In our proposed method, we use the lightweight ResNet18 as the backbone, from which its classifier is removed, and its layer 4 is employed as the subsequent part of FSCM in the identical-class object detector. For an anchor-based way, the RPN is derived from Faster R-CNN. All experiments are conducted on an NVIDIA 3090 GPU under Ubuntu 20.04.

The experiments use the Adam optimizer [58] with an initial learning rate of 1e-4. The batch size is set to 8, and the total number of epochs to 64. In the loss function, the parameters λ_1 and λ_2 from (11) are both set to 1 by default. Additionally, we implement a cosine annealing scheduler with a linear warm-up strategy to optimize the decay of the learning rate, ensuring smooth convergence during training. The proposed method and comparative experiments are all implemented under the PyTorch framework.

The proposed method, along with comparative experiments, is implemented using the PyTorch framework. During the training process, our approach takes a query image and a support object, along with its mask on the image, as inputs. This involves feature extraction and two rounds of feature comparison between the query image features and the support object features. The process ultimately outputs the location of objects that belong to the same category as the support target.

During the inference stage, our method serves as a subtask of the IS task by providing detected locations of identical-class objects as bounding box prompts. These prompts are used as inputs for the IS model to segment the same-category targets within the detected bounding box regions, thereby facilitating rapid IS. This capability significantly enhances the efficiency and accuracy of the IS processes.

C. Detection Performance Comparison

This section presents the detection performance comparisons on two datasets for different methods on the ICOD task.

1) Comparable Methods: We compared our proposed method with five recent FSOD methods [20], [52], [53], [59], [60] for the ICOD task. These FSOD methods can independently represent the support object by cropping the target region from the image for input into the backbone to obtain features. They can also be adjusted to operate without training on base classes, making them applicable to our proposed ICOD task.

FS_detection [20] introduces a reweighting module that maps support samples of a certain class to reweighting vectors, modulating the query image features to detect objects of the same class. DCNet [52] constructs a dense relation distillation (DRD) module that performs dense feature matching using support object features of certain classes to activate co-existing features in the input query. DAnA [53] transforms support images into query-position-aware features, guiding detection networks precisely by assigning customized support information to each local region of the query. FCT [59] builds a model based on a fully cross-transformer, incorporating cross-transformer into both the backbone and detection head to aggregate key information from both query and support images. DiGeo [60] employs a new training framework to learn geometry-aware features that enhance interclass separation and intraclass compactness for the FSOD task, addressing the issue of insufficient discriminative feature learning for all classes. These methods use support images of comparable size to query images and detect multiple categories simultaneously but cannot rely solely on the current category of the supporting image. Obviously, these methods differ somewhat from our task.

The latest FSOD methods, such as FM-FSOD [57] and SDDGR [61], are, however, not suitable for our work. FM-FSOD is based on the DETR framework and integrates large language models to achieve FSOD. SDDGR uses diffusion models to generate new samples based on a small amount of labeled data, including bounding boxes and category information, to train a robust object detector. These methods incorporate large language models and diffusion models, which are not currently applicable to our task, as they do not include text prompts and require consideration of real-time efficiency in interactions.

To adapt the recent five FSOD methods to our task, we made two adjustments: 1) support images are cropped from the target regions of the query image area and resized to 224 × 224, and 2) the network is configured to detect only objects of the same category as the support object.

2) On the EVLab Building Dataset: Our proposed method achieves the best results, with an AP of 40.97% and an F1 score of 66.37%, as shown in Table IV. Additionally, our method has a lower parameter count and a competitive inference speed of 40.66 FPS. Additionally, our method exhibits lower parameters and a competitive inference speed of

TABLE IV
COMPARISON OF DIFFERENT METHODS FOR THE TWO DATASETS. NOTE: "M" DENOTES MILLION, AND "FPS" STANDS FOR FRAMES PER SECOND

Methods	Params (M)	FPS	AP	Precision	Recall	F1-score
EVLab building dataset						
FS_Detection [20]	66.29	**180.93**	26.08	**76.83**	34.26	47.39
DCNet [52]	38.65	24.95	31.97	62.77	59.95	61.32
DAnA [53]	37.38	31.18	40.54	63.46	64.45	63.95
FCT [59]	24.39	15.70	39.12	61.55	62.61	62.08
DiGeo [60]	41.30	34.32	32.23	59.22	65.01	61.98
Ours	25.44	40.66	**40.97**	66.30	**66.43**	**66.37**
DOTA dataset						
FS_Detection [20]	66.29	**180.93**	17.41	45.15	20.84	28.51
DCNet [52]	38.65	24.95	16.81	45.24	44.74	44.99
DAnA [53]	37.38	31.18	18.17	42.06	43.32	42.68
FCT [59]	24.39	15.70	18.99	40.09	42.22	41.13
DiGeo [60]	41.30	34.32	19.01	45.49	44.99	45.24
Ours	25.44	40.66	**19.21**	**46.10**	**45.66**	**45.88**

40.66 FPS. In contrast, FS_detection, which uses a one-stage object detection approach, demonstrates the fastest inference speed but suffers from the lowest accuracy overall. Although FS_Detection achieves the highest precision of 76.8%, it has the lowest recall due to numerous missed detections among densely packed and small buildings, resulting in poor performance in both the AP and F1 score. FCT adopts a full transformer design, integrating support target and query image features from the feature extraction stage, potentially facing limitations due to simplified support targets. DiGeo, which considers interclass separation and intraclass compactness, performs poorly on single-class building datasets. We sequentially visualize the results of all methods across six regions, as depicted in Fig. 6. From the visualization, our method demonstrates the highest stability across various scenarios, followed by DCNet, which aligns with the F1 score and AP metrics. In the prediction results for Taiwan, Chongqing, and Xi'an regions, our method accurately predicts multiple buildings, while the ground truth represents these as a single large building, with several red boxes enclosed within a larger red box. This discrepancy primarily arises from inconsistencies and a lack of precision in the manual annotations of our ground truth samples. In spite of these issues, our proposed method accurately separates these buildings, achieving the best performance.

3) On the DOTA Dataset: We conduct further validation of various methods on the multiclass DOTA dataset. As shown in Table IV, our proposed method achieves the best performance, with an AP of 19.21% and an F1 score of 45.88%, followed by DiGeo. DiGeo performs well on the multiclass DOTA dataset due to its focus on interclass separation and intraclass compactness. FS_Detection shows the lowest performance in terms of F1 score, while DCNet performs poorly in terms of AP, indicating that FS_Detection requires fine-tuning and is heavily dependent on parameter adjustments. The query results for different methods are visualized in Fig. 7, demonstrating that our method outperforms the others.

Our method excels particularly with large objects, such as tennis courts, planes, harbors, basketball courts, and baseball diamonds, with DAnA as the next best performer. DiGeo and FCT, however, perform the worst on baseball diamond detection, likely due to the scarcity and imbalance of samples in this category. For smaller, densely packed targets like ships, DAnA and FCT miss many detections, while our method has a few FPs due to a fixed NMS threshold. Notably, in detecting planes, ships, large vehicles, and baseball diamonds, FS_Detection, DCNet, and DiGeo detect objects of other categories, whereas our method consistently detects only targets of the same category as the support object. FCT struggles with detecting small, gray storage tanks due to the support object being larger and whiter, whereas our method performs well by considering semantic and spatial similarity. The key to our approach lies in querying only targets of the same category as the support targets. Overall, our approach demonstrates superior performance on the multiclass ICOD task.

D. Ablation Study

We conduct a series of experiments on the EVLab Building Dataset and the DOTA dataset to validate the importance of each component of the proposed method. Additionally, we examine the influence of different backbones on the performance of the proposed method. The impact of using the mask of the support object is, furthermore, validated within the designed S3QFM. Finally, we investigate various factors influencing the performance of the proposed method, such as the number of expanded pixels for the bounding box of the support object and the number of support objects.

1) Impact of Each Component of Our Method: As shown in Table V, we conduct experiments on different components of our method on two datasets. As a baseline, we use two convolutional layers for feature fusion to replace S3QFM. In this setup, the features of the support objects are resized to match the dimensions of the query image features, and the two

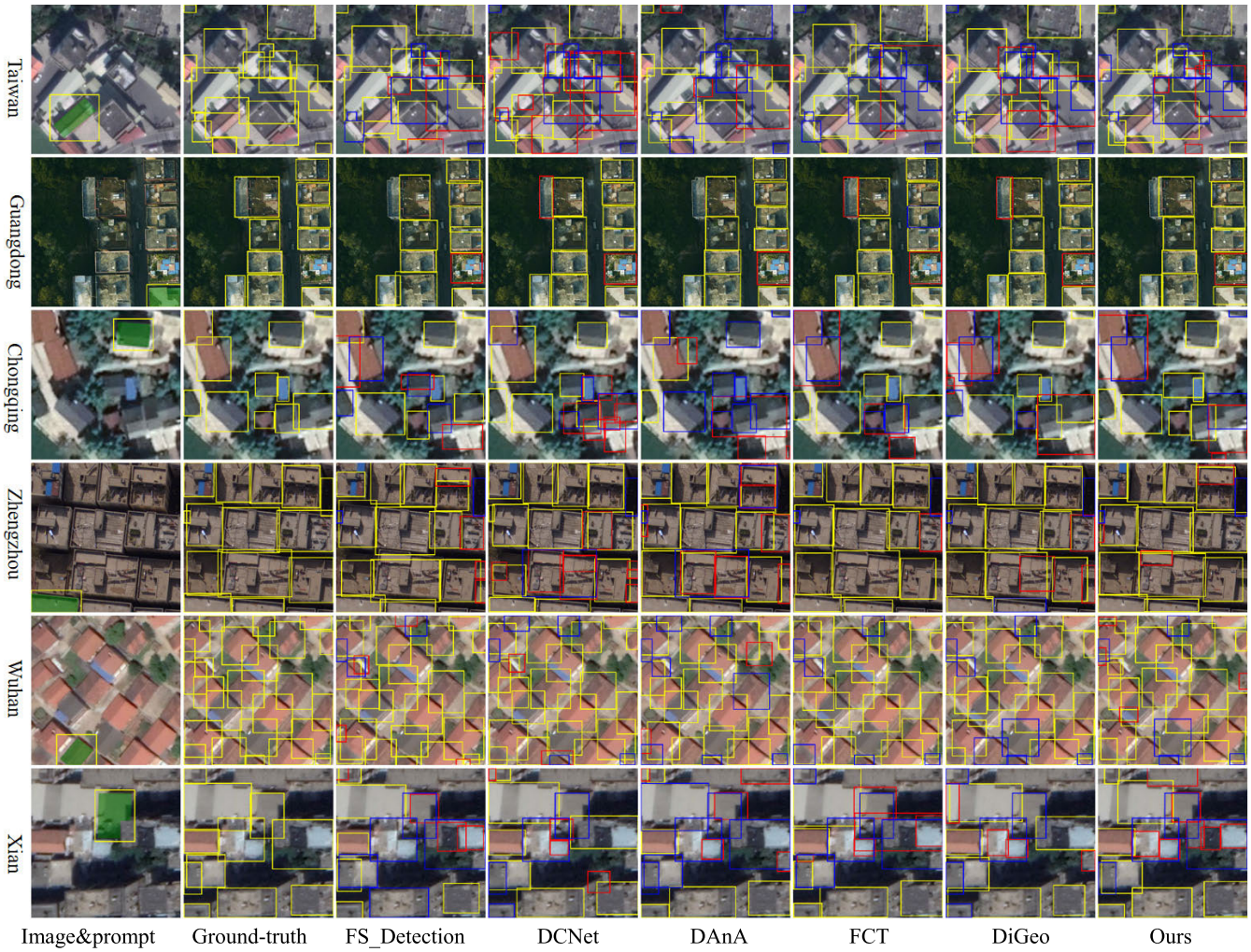


Fig. 6. Visualization of different methods on the EVLab building dataset. Note: in the prediction results, yellow boxes indicate TPs, red boxes indicate FPs, and blue boxes indicate FNs.

features are concatenated. On the EVLab building dataset, the baseline achieves an AP of 37.88% and an F1 score of 63.56%. After incorporating Se3M, the AP increases by about 1%, and the F1 score increases by 1.29%. The baseline using Sp3M shows more significant improvements, with AP and F1 scores increasing by 2.0% and 1.77%, respectively. The performance is further improved when Se3M and Sp3M are used together, called S3QFM. After adding FCSM, the method achieves the best results.

Given that the EVLab building dataset contains only a single category of buildings, we continue to validate the components of our proposed method on the DOTA dataset. Quantitative results indicate that the performance of the method is gradually improved. Notably, baseline using Sp3M did not demonstrate a significant advantage over baseline using Se3M on multiclass DOTA dataset. This is partly because semantic features significantly influence the process of searching for features similar between target features and queried image features. The organized DOTA dataset includes 12 categories, which, furthermore, increases the complexity of the task. In the multiclass DOTA dataset, our introduced FCSM shows an improvement of 0.78% in AP and 1.35% in F1 score, demonstrating a substantial improvement compared to the single-building task.

2) Impact of Using Different Backbones: We evaluate the performance of the proposed ICODet method using different backbones on two datasets, as shown in Table VI. The backbones used include ResNet50 [26], ResNet101 [26], VGG16 [62], the recent ConvNeXt V2 [63], and Dinov2 [64]. ResNet50 and ResNet101 are chosen to examine the impact of increasing parameters on model performance. VGG16 represents a different CNN architecture, while ConvNeXt V2 is a recent and powerful model. Additionally, Dinov2, based on the transformer architecture, is noted for its strong generalization capabilities. Experiments were conducted on the EVLab building dataset and the DOTA dataset.

As observed in Table VI, the detection performance for identical-class objects progressively improves with ResNet50 and ResNet101 backbones, with ResNet101 achieving better results and an inference speed of 10.24 images per second (for images sized 512 $\times$ 512). ICODet's performance peaks when using ConvNeXt V2 as the backbone due to its robust architectural design; however, the complexity of ConvNeXt

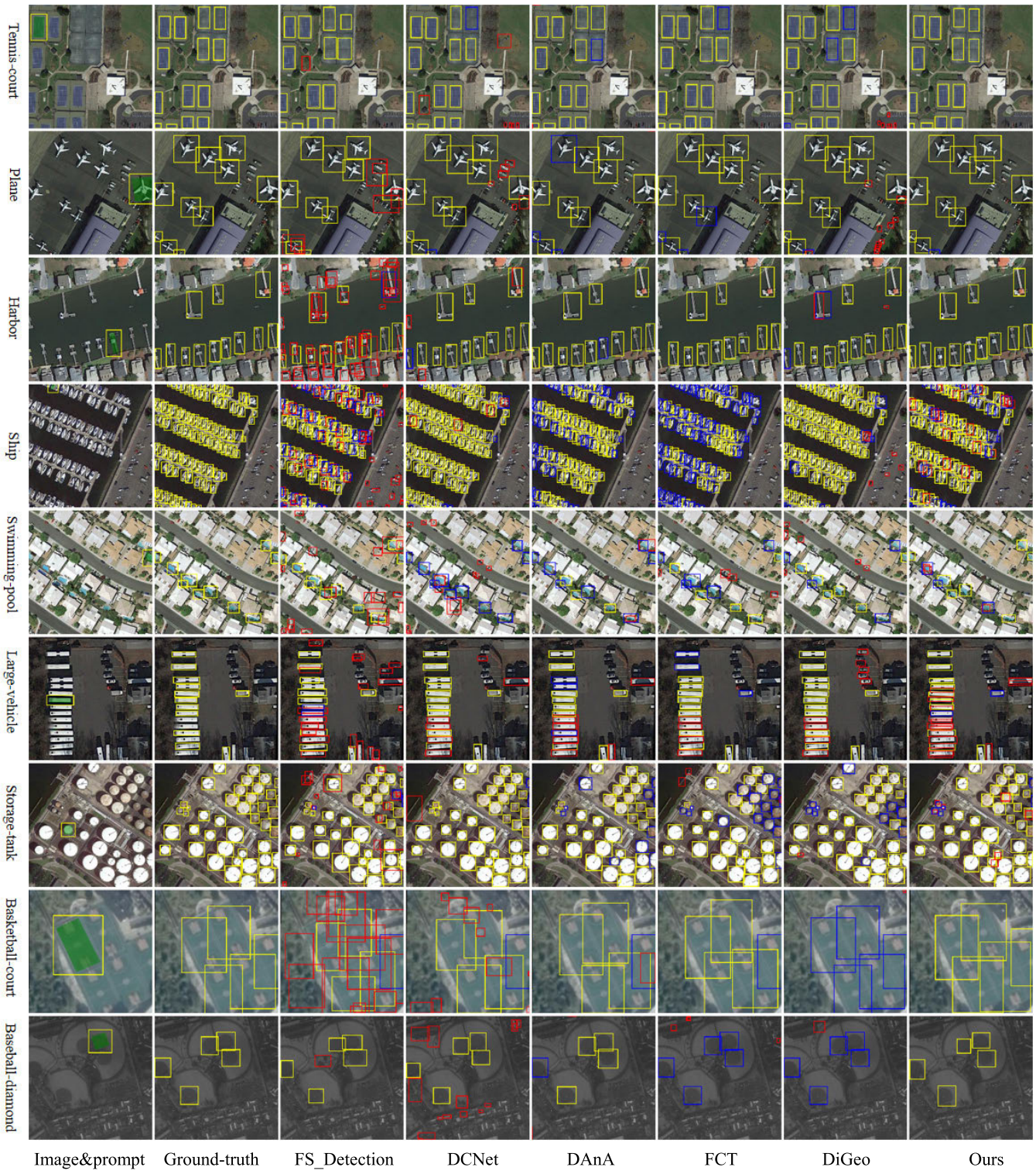


Fig. 7. Visualization of different methods on the DOTA dataset. Note: in the prediction results, yellow boxes indicate TPs, red boxes indicate FPs, and blue boxes indicate FNs.

V2 results in a lower inference speed of 9.49 images per second. Although VGG16 has a comparable parameter count to ConvNeXt V2, its detection performance is lower. The performance with Dinov2 is the poorest, likely because our designed S3QFM is primarily CNN-based and does not incorporate transformer structures, making it challenging to adapt to the transformer-based Dinov2.

While increasing model parameters can enhance the performance of the ICOD task, it is essential to consider the need for real-time interaction in practical applications. We have, therefore, chosen ResNet18 as the backbone for the current version. In future work, we plan to use models with higher parameters to further improve detection capabilities.

TABLE V
ABLATION OF OUR PROPOSED METHOD ON THE TWO DATASETS. "M" STANDS FOR MILLION AND "G" REPRESENTS BILLION

Methods	Params (M)	Flops (G)	AP	Precision	Recall	F1-score
EVLab building dataset						
baseline	13.23	257.27	37.88	58.42	**69.68**	63.56
baseline+Se3M	14.87	259.45	38.86	64.50	65.21	64.85
baseline+Sp3M	22.74	263.39	39.89	65.62	65.04	65.33
baseline +S3QFM	24.52	265.21	40.82	**66.80**	64.80	65.79
baseline +S3QFM+ FCSM	25.44	373.35	**40.97**	66.30	66.43	**66.37**
DOTA dataset						
baseline	13.23	257.27	15.65	37.48	43.65	40.33
baseline+Se3M	14.87	259.45	17.98	43.17	44.12	43.64
baseline+Sp3M	22.74	263.39	18.31	43.89	44.04	43.96
baseline +S3QFM	24.52	265.21	18.43	45.24	43.85	44.53
baseline +S3QFM+ FCSM	25.44	373.35	**19.21**	**46.10**	**45.66**	**45.88**

TABLE VI
PERFORMANCE OF ICODET USING DIFFERENT BACKBONES ON THE TWO DATASETS

Backbone	Params (M)	FPS	AP	Precision	Recall	F1-score
EVLab building dataset						
ResNet18	25.44	**40.66**	40.97	66.30	66.43	66.37
ResNet50	87.83	12.01	42.18	68.57	65.99	67.26
ResNet101	106.82	10.24	43.09	72.32	66.00	69.02
VGG16	190.60	28.72	39.47	71.51	62.32	66.60
ConvNeXt V2	144.00	9.49	**49.03**	**77.47**	**71.96**	**74.61**
Dinov2	147.99	19.01	34.38	68.94	58.37	63.22
DOTA dataset						
ResNet18	25.44	**40.66**	19.21	46.10	45.66	45.88
ResNet50	87.83	12.01	20.26	48.62	45.47	47.00
ResNet101	106.82	10.24	20.94	48.74	45.83	47.24
VGG16	190.60	28.72	20.04	49.50	46.16	47.77
ConvNeXt V2	144.00	9.49	**24.56**	**54.50**	**52.64**	**53.55**
Dinov2	147.99	19.01	16.36	43.03	38.64	40.72

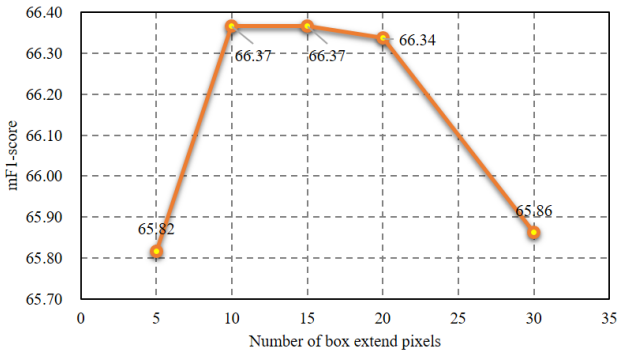


Fig. 8. Influence of the number of expanded pixels of the external rectangular box.

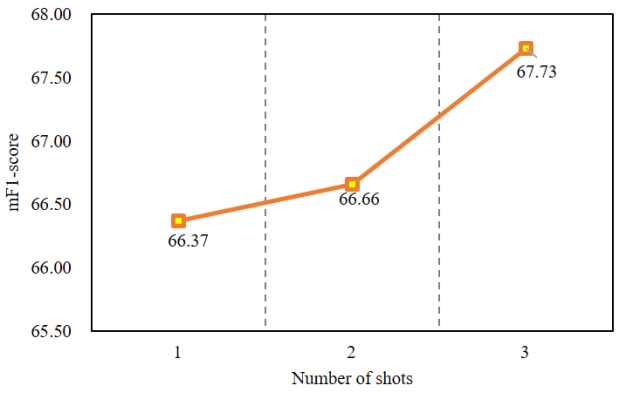
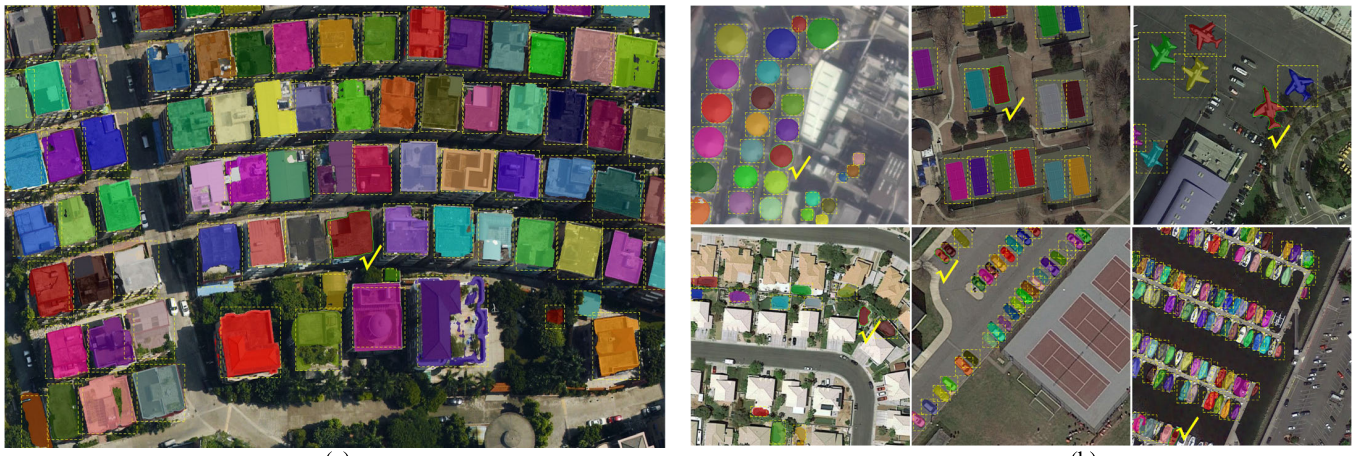


Fig. 9. Impact of the number of the support objects on the EVLab building dataset.

3) Impact of Using the Mask of the Support Target: In our proposed ICODet, we use the mask of the support object in the S3QFM to leverage the background and semantic information surrounding the support object. As shown in Table VII, we evaluate the impact of masks within the S3QFM.



(a) (b)

Fig. 10. Application of fast IS using our proposed ICODet and an IS model. The yellow checkmark (√) denotes the point prompts used by the IS model to generate the mask of the support object. The yellow dashed box illustrates the detection results of ICODet, while the masks in various colors indicate the segmentation results obtained through the IS model. (a) Building extraction in high-resolution RSI. (b) Some typical remote sensing object segmentation.

When neither Se3M NOR Sp3M uses masks, the performance degradation of our proposed method is greatest. After implementing masks separately in Se3M and Sp3M, there are some improvements. Particularly, the best results are achieved when both used masks simultaneously.

4) Impact of the Extended Pixels of the Box: Our proposed task is designed based on the image of a specific region, where the supporting object comes from the current image; therefore, we need to crop these objects as support samples. In this process, the extent to which the bounding box of the support targets is expanded outward is a crucial factor; therefore, we conduct experiments to expand the bounding box of the target by N pixels on each side (top, bottom, left, right), $N \in \{5, 10, 15, 20, 30\}$, as shown in Fig. 8. The experimental results show that when the expansion size is set to 10 or 15 pixels. The performance reaches the peak. Continuing to increase the number of extended pixels beyond this point will cause performance degradation. This drop is most likely due to the inclusion of parts of other buildings as a background in the expanded scope, resulting in confusion for the proposed method. In this article, we set the box expansion to 10 pixels for all comparative experiments.

5) Impact of the Number of Support Objects: For our proposed ICOD task, the selection and number of support targets influence the model's ability to identify more of the same type of targets. To this end, we validate the impact of randomly choosing 1–3 support targets on the performance of the proposed method. As shown in Fig. 9, the performance improves with an increase in the number of support targets. When the number of support targets reaches or exceeds two, the features from multiple support targets are averaged in the S3QFM, and these averaged features are then used for similarity calculations. This approach allows the averaged features to better capture the overall characteristics of the buildings in the current query image. This finding confirms the effectiveness of our approach for the ICOD task. In our experimental setup, we have set the number of support objects to one to streamline the process while maintaining robustness.

E. Application

We develop a fast IS application using our proposed ICODet and an IS model for remote sensing images as illustrated in Fig. 1. This application initially performs IS on a support target by using point prompts or boxes to obtain the support target's mask. It then uses our ICODet to heuristically identify identical-class objects within the image. Finally, the application employs the IS model and the detected bounding boxes to segment the identical-class objects. As demonstrated in Fig. 10, the application effectively recognizes and segments all buildings after interactively segmenting one building. Most buildings are segmented with high quality, though a few present challenges due to vegetation or other structures on their rooftops. Additionally, the application performs well in quickly segmenting storage tanks, swimming pools, planes, and small vehicles in high-resolution images. While ships in dense scenes result in a few missed detections due to their small size and low image resolution, the IS results for most categories still meet the requirements. The IS model used is derived from SAM [16]. In future work, we plan to fine-tune this model to better adapt it to our remote sensing targets, such as complex buildings and smaller-sized objects.

V. DISCUSSION

The proposed ICODet incorporates a S3QFM that learns similarity features between support object features and query image features from both semantic and spatial perspectives. This approach has proven effective for both single-category and multicategory datasets; however, it still falls short of the demands for rapid IS. Our method struggles with small targets, such as ships, and with poor distinguishability between categories, such as helicopters and airplanes or large and small vehicles, leading to lower accuracy on the DOTA dataset. In future work, we plan to incorporate intraclass variability into the transformer-based DETR structure to enhance its ability to detect small targets and improve the distinguishability between subcategories. Additionally, the S3QFM is primarily designed based on CNN; we will redesign it as

TABLE VII
EFFECTS OF USING MASK AT DIFFERENT POSITIONS OF OUR METHOD ON EVLAB BUILDING DATASET

Methods	Params (M)	Flops (G)	AP	Precision	Recall	F1-score
Without-mask	23.33	264.97	39.65	64.44	65.82	65.12
Se3M - mask	23.34	264.98	39.73	65.80	64.80	65.29
Sp3M - mask	24.52	265.21	39.81	66.30	65.02	65.65
With masks	25.44	373.35	**40.97**	**66.30**	**66.43**	**66.37**

a transformer-based module to better leverage the powerful feature representation capabilities of transformers.

Our proposed ICODet, furthermore, effectively detects identical-class objects in fast IS applications; however, it struggles with complex architectural rooftops when using the SAM. We intend to fine-tune SAM to enhance its performance for our fast IS tasks in the near future.

VI. CONCLUSION

In this research, a new ICOD task and an identical-class object detection network denoted as ICODet, are proposed for faster IS for high-resolution remote sensing images. The ICOD task is designed to identify only those objects that belong to the same class as the support target while excluding objects from other categories. Our proposed method adopts a two-stage object detection design, using a lightweight feature extractor to capture features from both the query image and the supporting target category. By constructing a feature similarity analysis module that incorporates semantic similarity and spatial search, we analyze the similarity between support object features and query image features at both the feature-space and semantic levels. A simple yet effective comparison of the preliminary detection results with the query targets, furthermore, enhances detection performance in multiclass tasks. The video demonstrating the effects of faster IS, along with the publicly available EVLab Building dataset and the reorganized DOTA dataset, can be found at https://github.com/zhilyzhang/ICODet. This work greatly improves the efficiency of sample annotations, offering significant practical value for data production and fundamental applications in the remote sensing community.

REFERENCES

[1] D. Li, M. Wang, and J. Jiang, "China's high-resolution optical remote sensing satellites and their mapping applications," *Geo-spat. Inf. Sci.*, vol. 24, no. 1, pp. 85–94, 2021.

[2] K. E. Sawaya, L. G. Olmanson, N. J. Heinert, P. L. Brezonik, and M. E. Bauer, "Extending satellite remote sensing to local scales: Land and water resource monitoring using high-resolution imagery," *Remote Sens. Environ.*, vol. 88, nos. 1–2, pp. 144–156, Nov. 2003.

[3] C. Jun, Y. Ban, and S. Li, "Open access to Earth land-cover map," *Nature*, vol. 514, no. 7523, p. 434, Oct. 2014.

[4] Y. Cao and Q. Weng, "A deep learning-based super-resolution method for building height estimation at 2.5 m spatial resolution in the northern hemisphere," *Remote Sens. Environ.*, vol. 310, Aug. 2024, Art. no. 114241.

[5] N. Kussul, M. Lavreniuk, S. Skakun, and A. Shelestov, "Deep learning classification of land cover and crop types using remote sensing data," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 778–782, May 2017.

[6] K. E. Joyce, S. E. Belliss, S. V. Samsonov, S. J. McNeill, and P. J. Glassey, "A review of the status of satellite remote sensing and image processing techniques for mapping natural hazards and disasters," *Prog. Phys. Geography, Earth Environ.*, vol. 33, no. 2, pp. 183–207, Apr. 2009.

[7] Y. Cao and X. Huang, "A full-level fused cross-task transfer learning method for building change detection using noise-robust pretrained networks on crowdsourced labels," *Remote Sens. Environ.*, vol. 284, Jan. 2023, Art. no. 113371.

[8] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2018.

[9] G.-S. Xia et al., "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983.

[10] Z. Zhang, Q. Zhang, X. Hu, M. Zhang, and D. Zhu, "On the automatic quality assessment of annotated sample data for object extraction from remote sensing imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 201, pp. 153–173, Jul. 2023.

[11] J. He, C.-S. Kim, and C.-C. J. Kuo, *Interactive Segmentation Techniques: Algorithms and Performance Evaluation*. Cham, Switzerland: Springer, 2014.

[12] X. Chen, Z. Zhao, Y. Zhang, M. Duan, D. Qi, and H. Zhao, "FocalClick: Towards practical interactive image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1290–1299.

[13] Z. Lin, Z.-P. Duan, Z. Zhang, C.-L. Guo, and M.-M. Cheng, "FocusCut: Diving into a focus view in interactive segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2627–2636.

[14] K. Sofiiuk, I. A. Petrov, and A. Konushin, "Reviving iterative training with mask guidance for interactive segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 3141–3145.

[15] H. You et al., "InterFormer real-time interactive image segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2023, pp. 22301–22311.

[16] A. M. Kirillov et al., "Segment anything," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2023, pp. 4015–4026.

[17] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," in *Proc. 14th Eur. Conf. Comput. Vision (ECCV)*, Amsterdam, The Netherlands. Cham, Switzerland: Springer, 2016, pp. 241–257.

[18] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surveys*, vol. 40, no. 2, pp. 1–60, Apr. 2008.

[19] S. Antonelli et al., "Few-shot object detection: A survey," *ACM Comput. Surv.*, vol. 54, no. 11, pp. 1–37, 2022.

[20] B. Kang, Z. Liu, X. Wang, F. Yu, J. Feng, and T. Darrell, "Few-shot object detection via feature reweighting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8419–8428.

[21] E. Lu, W. Xie, and A. Zisserman, "Class-agnostic counting," in *Proc. Asian Conf. Comput. Vis. (ACCV)*. Cham, Switzerland: Springer, 2018, pp. 669–684.

[22] S.-D. Yang, H.-T. Su, W. H. Hsu, and W.-C. Chen, "Class-agnostic few-shot object counting," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 869–877.

[23] T.-I. Hsieh, Y. Lo, H.-T. Chen, and T.-L. Liu, "One-shot object detection with co-attention and co-excitation," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2019, pp. 1–12.

[24] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, Feb. 2016, pp. 658–666.

[25] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, Dec. 2015, pp. 91–99.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[27] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.

[28] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman, "Geodesic star convexity for interactive image segmentation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3129–3136.

[29] C. Rother, V. Kolmogorov, and A. Blake, "'GrabCut'—Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[30] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. 8th IEEE Int. Conf. Comput. Vision. ICCV*, vol. 1, Aug. 2001, pp. 105–112.

[31] N. Xu, B. Price, S. Cohen, S. Yan, and T. S. Huang, "Deep interactive object selection," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jun. 2016, pp. 373–381.

[32] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool, "Deep extreme cut: From extreme points to object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 616–625.

[33] Z. Lin, Z. Zhang, L.-Z. Chen, M.-M. Cheng, and S.-P. Lu, "Interactive image segmentation with first click attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13336–13345.

[34] K. Sofiiuk, I. Petrov, O. Barinova, and A. Konushin, "F-BRS: Rethinking backpropagating refinement for interactive segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8620–8629.

[35] Z. Shu, X. Hu, and H. Dai, "Progress guidance representation for robust interactive extraction of buildings from remotely sensed images," *Remote Sens.*, vol. 13, no. 24, p. 5111, Dec. 2021.

[36] X. Zou et al., "Segment everything everywhere all at once," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2023, pp. 1–11.

[37] M. Zhou et al., "Interactive segmentation as Gaussian process classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 19488–19497.

[38] F. Li et al., "Visual in-context prompting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 12861–12871.

[39] J. Lu, J. Hu, and J. Zhou, "Deep metric learning for visual understanding: An overview of recent advances," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 76–84, Nov. 2017.

[40] A. Fathi et al., "Semantic instance segmentation via deep metric learning," 2017, *arXiv:1703.10277*.

[41] X. Jiang, N. Zhou, and X. Li, "Few-shot segmentation of remote sensing images using deep metric learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[42] M. Kaya and H. Ş. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, p. 1066, Aug. 2019.

[43] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5149–5169, Sep. 2021.

[44] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surveys*, vol. 53, no. 3, pp. 1–34, Jun. 2020.

[45] N. Dong and E. P. Xing, "Few-shot semantic segmentation with prototype learning," in *Proc. BMVC*, Jan. 2018, pp. 1–13.

[46] V. Ranjan, U. Sharma, T. Nguyen, and M. Hoai, "Learning to count everything," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3393–3402.

[47] M. Shi, H. Lu, C. Feng, C. Liu, and Z. Cao, "Represent, compare, and learn: A similarity-aware framework for class-agnostic counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9519–9528.

[48] K. T. Ahmed, S. Ummesafi, and A. Iqbal, "Content based image retrieval using image features information fusion," *Inf. Fusion*, vol. 51, pp. 76–99, Nov. 2019.

[49] S. R. Dubey, "A decade survey of content based image retrieval using deep learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 5, pp. 2687–2704, May 2021.

[50] Y. Li, J. Ma, and Y. Zhang, "Image retrieval from remote sensing big data: A survey," *Inf. Fusion*, vol. 67, pp. 94–115, Mar. 2021.

[51] G. Zhang, Z. Luo, K. Cui, S. Lu, and E. P. Xing, "Meta-DETR: Image-level few-shot detection with inter-class correlation exploitation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 12832–12843, Nov. 2022.

[52] H. Hu, S. Bai, A. Li, J. Cui, and L. Wang, "Dense relation distillation with context-aware aggregation for few-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10180–10189.

[53] T.-I. Chen et al., "Dual-awareness attention for few-shot object detection," *IEEE Trans. Multimedia*, vol. 25, pp. 291–301, 2021.

[54] B. Yan, C. Lang, G. Cheng, and J. Han, "Understanding negative proposals in generic few-shot object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 7, pp. 5818–5829, Jul. 2024.

[55] Y. Liu et al., "Few-shot object detection in remote-sensing images via label-consistent classifier and gradual regression," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5612114.

[56] J. Han, Y. Ren, J. Ding, K. Yan, and G.-S. Xia, "Few-shot object detection via variational feature aggregation," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2023, vol. 37, no. 1, pp. 755–763.

[57] G. Han and S.-N. Lim, "Few-shot object detection with foundation models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 28608–28618.

[58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[59] G. Han, J. Ma, S. Huang, L. Chen, and S.-F. Chang, "Few-shot object detection with fully cross-transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5311–5320.

[60] J. Ma, Y. Niu, J. Xu, S. Huang, G. Han, and S.-F. Chang, "DiGeo: Discriminative geometry-aware learning for generalized few-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 3208–3218.

[61] J. Kim, H. Cho, J. Kim, Y. Y. Tiruneh, and S. Baek, "SDDGR: Stable diffusion-based deep generative replay for class incremental object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 28772–28781.

[62] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[63] S. Woo et al., "ConvNeXt v2: Co-designing and scaling ConvNets with masked autoencoders," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 16133–16142.

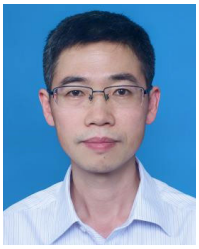
[64] M. Oquab et al., "Dinov2: Learning robust visual features without supervision," 2023, *arXiv:2304.07193*.

Zhili Zhang (Associate Member, IEEE) received the B.S. degree from the School of Geosciences and Info-Physics, Central South University, Changsha, China, in 2018. He is currently pursuing the Ph.D. degree with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China.

His research interests mainly include deep learning, human-computer interaction, and remote sensing image processing.

Jiabo Xu (Student Member, IEEE) received the B.S. degree in software engineering from Nanchang Hangkong University, Nanchang, China, in 2019, and the M.S. degree in computer technologies from the School of Mathematics and Computer Sciences, Nanchang University, Nanchang, in 2023. He is currently pursuing the Ph.D. degree in photogrammetry and remote sensing with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China.

His research interests include computer vision and point cloud processing.



Xiangyun Hu received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2001.

From 2002 to 2005, he was a Post-Doctoral Research Fellow with the Department of Earth and Space Science and Engineering, Lassonde School of Engineering, York University, Toronto, ON, Canada. He has developed a feature extraction technology SmartDigitizer acquired by PCI Geomatics, Markham, ON, Canada; Leica Geosystems, Norcross, GA, USA; and Microsoft, Redmond, WA, USA. From 2005 to 2010, he was a Senior Software Engineer with ERDAS Inc., Norcross. He is currently a Professor and the Head of the Department of Photogrammetry, School of Remote Sensing and Information Engineering, Wuhan University. He is also an Adjunct Professor with Hubei Luojia Laboratory, Wuhan. Recently, he has been leading a team in developing an open source deep learning framework—LuojiaNET. He has published more than 100 articles in journals and conferences on intelligent feature extraction of remotely sensed data.

Bingnan Yang received the B.S. degree from Nanjing Normal University, Nanjing, China, and the M.Sc. degree from The University of Manchester, Manchester, U.K. He is currently pursuing the Ph.D. degree with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China.

His research interests include remote sensing image processing and deep learning.

Mi Zhang received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2018.

He is currently an Associate Researcher at Wuhan University. He conducted Postdoctoral Research Fellow coadvised by Academician Jiancheng Li and Jianya Gong at the School of Remote Sensing and Information Engineering. Since 2018, he has been the Chief Artificial Intelligence working scientist at Handleray Corporation, San Diego, CA, USA. His research interests mainly include semantic segmentation, intelligent remote sensing image processing, and machine learning systems, with particular interest in semantic object segmentation and artificial intelligence systems for remote sensing.

Dr. Zhang is a reviewer of CVPR'2018, ICCV'2019, AAAI'2020, CVPR'2021, ICCV'2021, a Program Committee Member, and a reviewer of more than ten journals.